ANALYTIX
AUDIT

20/04/2022

PREMIUM AUDIT

NAME OF PROJECT	Solana Dao
TOKEN TESTNET	0x234EDaF20BD60f246bd926ed8547837D75141988
WEBSITE / TELEGRAM	https://solanadao.link/

DISCLAIMER: PLEASE READ FULL AUDIT

IMPORTANT DISCLAIMER

I.

NOT RESPONSIBLE

Analytix Audit holds no responsibility for any actions from the project in this audit.

II.

NOT GUARANTEE

Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.

III.

INFORMATION

Analytix Audit researches and provides public information about the project in an easy-to-understand format for the common person.

IV.

AUDIT AGREEMENT

This audit agreement does not guarantee ANY illicit actions by the project team and does not serve as an advocation for the project.

V.

NO ADVICE

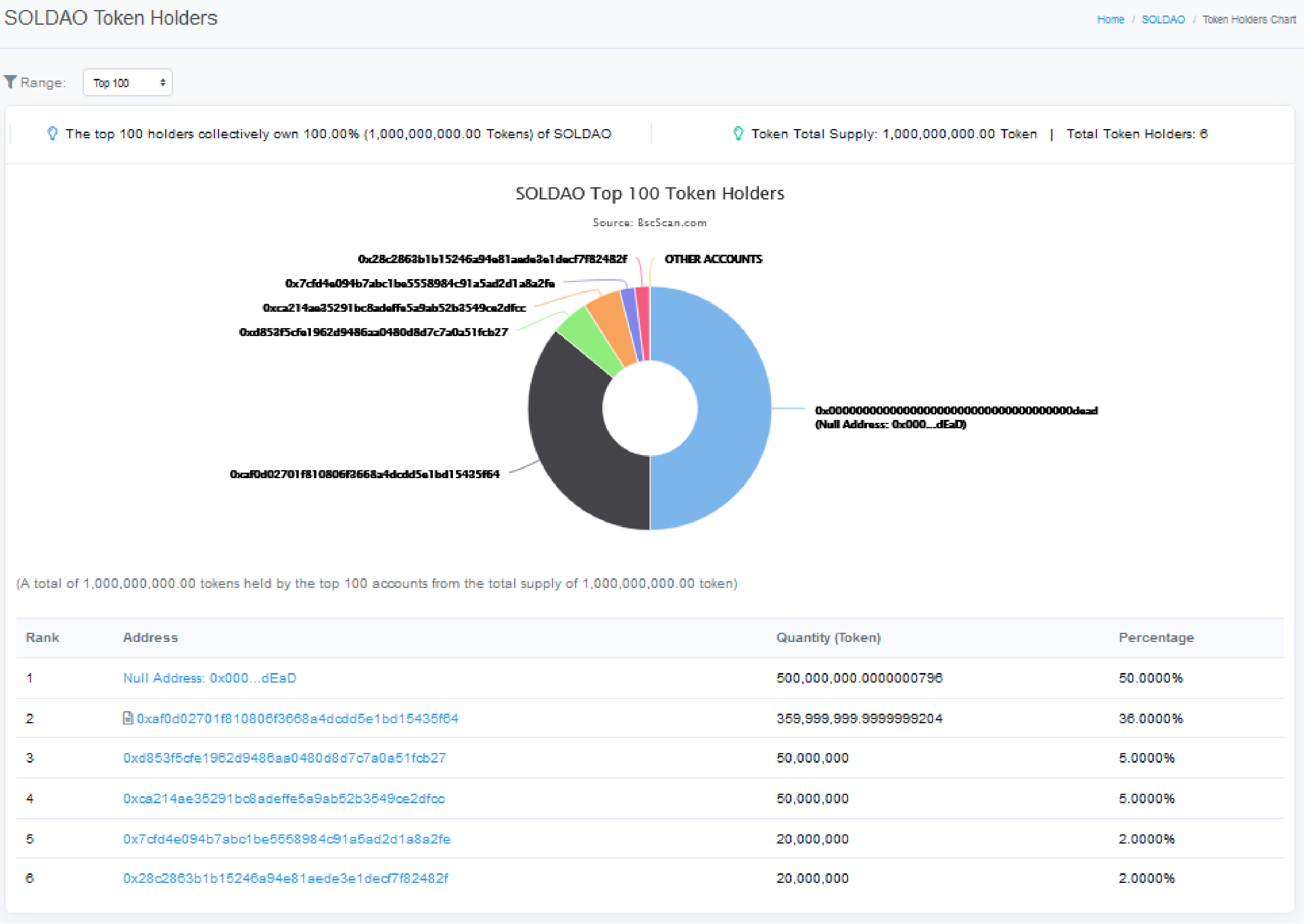
Analytix Audit in no way takes responsibility for any losses, not does Analytix Audit encourage any speculative investments.

VI.

DISCLAIMER

The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

TOKENOMICS - OVERVIEW



SOLDAO Token Holders

Home / SOLDAO / Token Holders Chart

Range: Top 100

The top 100 holders collectively own 100.00% (1,000,000,000.00 Tokens) of SOLDAO

Token Total Supply: 1,000,000,000.00 Token | Total Token Holders: 6

SOLDAO Top 100 Token Holders

Source: BscScan.com

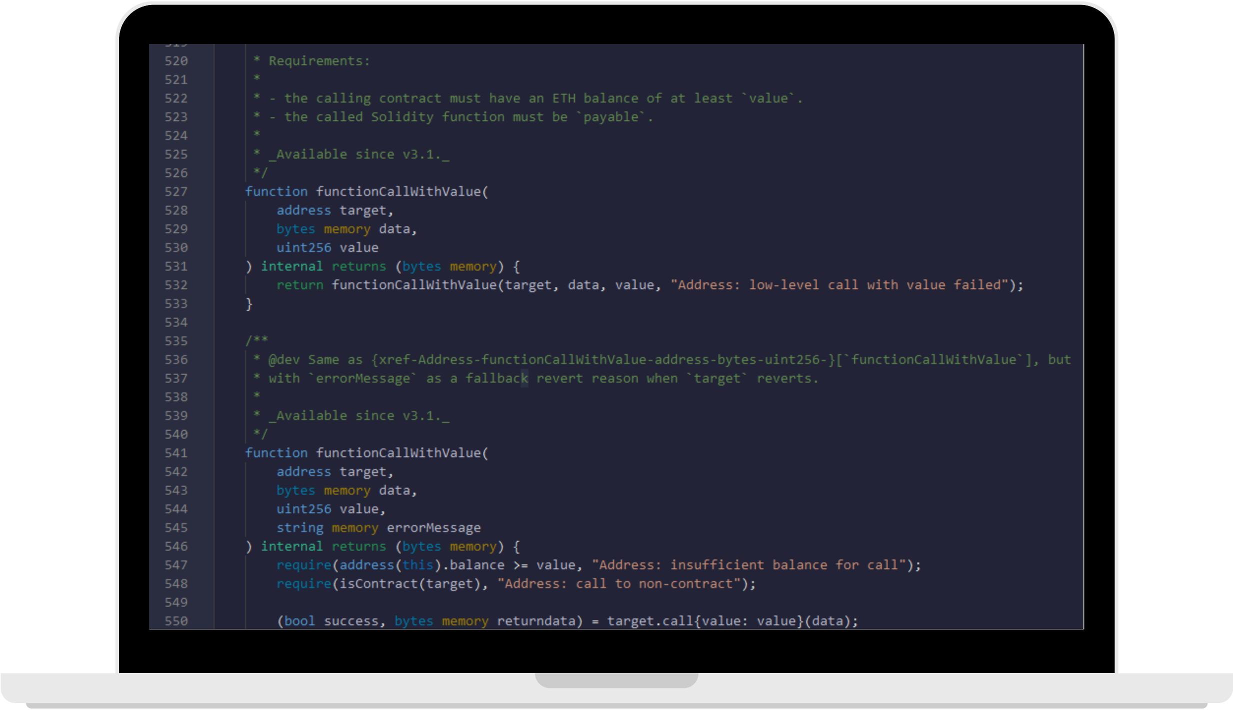
0x28c2863b1b15246a94e81aede3e1decf7f82482f
OTHER ACCOUNTS
0x7cfd4e094b7abc1be5558984c91a5ad2d1a8a2fe
0xca214ae35291bc8adeffe5a9ab52b3549ce2dfcc
0xd853f5cfe1962d9486aa0480d8d7c7a0a51fcb27
0x000000000000000000000000000000000000dead
(Null Address: 0x000...dEaD)
0xaf0d02701f810806f3668a4dcdd5e1bd15435f64

(A total of 1,000,000,000.00 tokens held by the top 100 accounts from the total supply of 1,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	Null Address: 0x000...dEaD	500,000,000.0000000796	50.0000%
2	0xaf0d02701f810806f3668a4dcdd5e1bd15435f64	359,999,999.9999999204	36.0000%
3	0xd853f5cfe1962d9486aa0480d8d7c7a0a51fcb27	50,000,000	5.0000%
4	0xca214ae35291bc8adeffe5a9ab52b3549ce2dfcc	50,000,000	5.0000%
5	0x7cfd4e094b7abc1be5558984c91a5ad2d1a8a2fe	20,000,000	2.0000%
6	0x28c2863b1b15246a94e81aede3e1decf7f82482f	20,000,000	2.0000%

(This comment refers to when the audit was carried out maybe team locked tokens later)

CONTRACT CODE - OVERVIEW

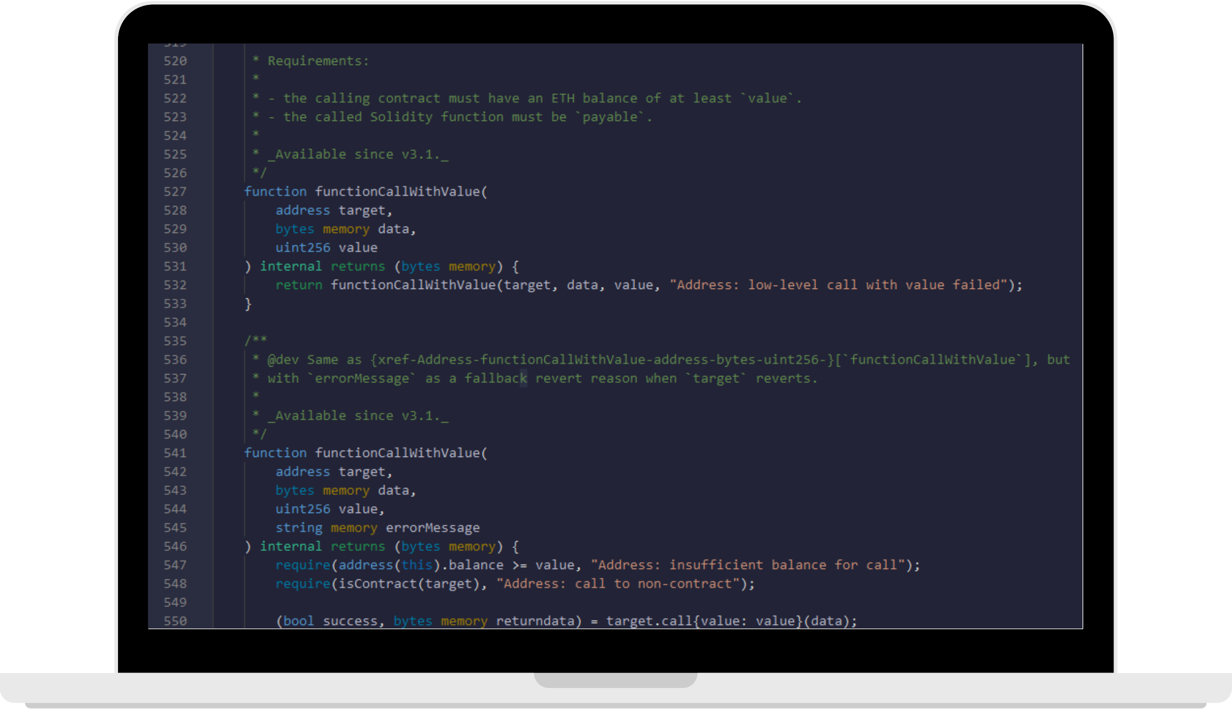


```
     * Requirements:
     *
     * - the calling contract must have an ETH balance of at least `value`.
     * - the called Solidity function must be `payable`.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value
    ) internal returns (bytes memory) {
        return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
    }

    /**
     * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
     * with `errorMessage` as a fallback revert reason when `target` reverts.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value,
        string memory errorMessage
    ) internal returns (bytes memory) {
        require(address(this).balance >= value, "Address: insufficient balance for call");
        require(isContract(target), "Address: call to non-contract");

        (bool success, bytes memory returndata) = target.call{value: value}(data);
```

QUICK REVIEW

- Owner can change fees up to 100%
- Owner can't set Tx Limits.
- Owner can withdraw funds from wallet / smart contracts

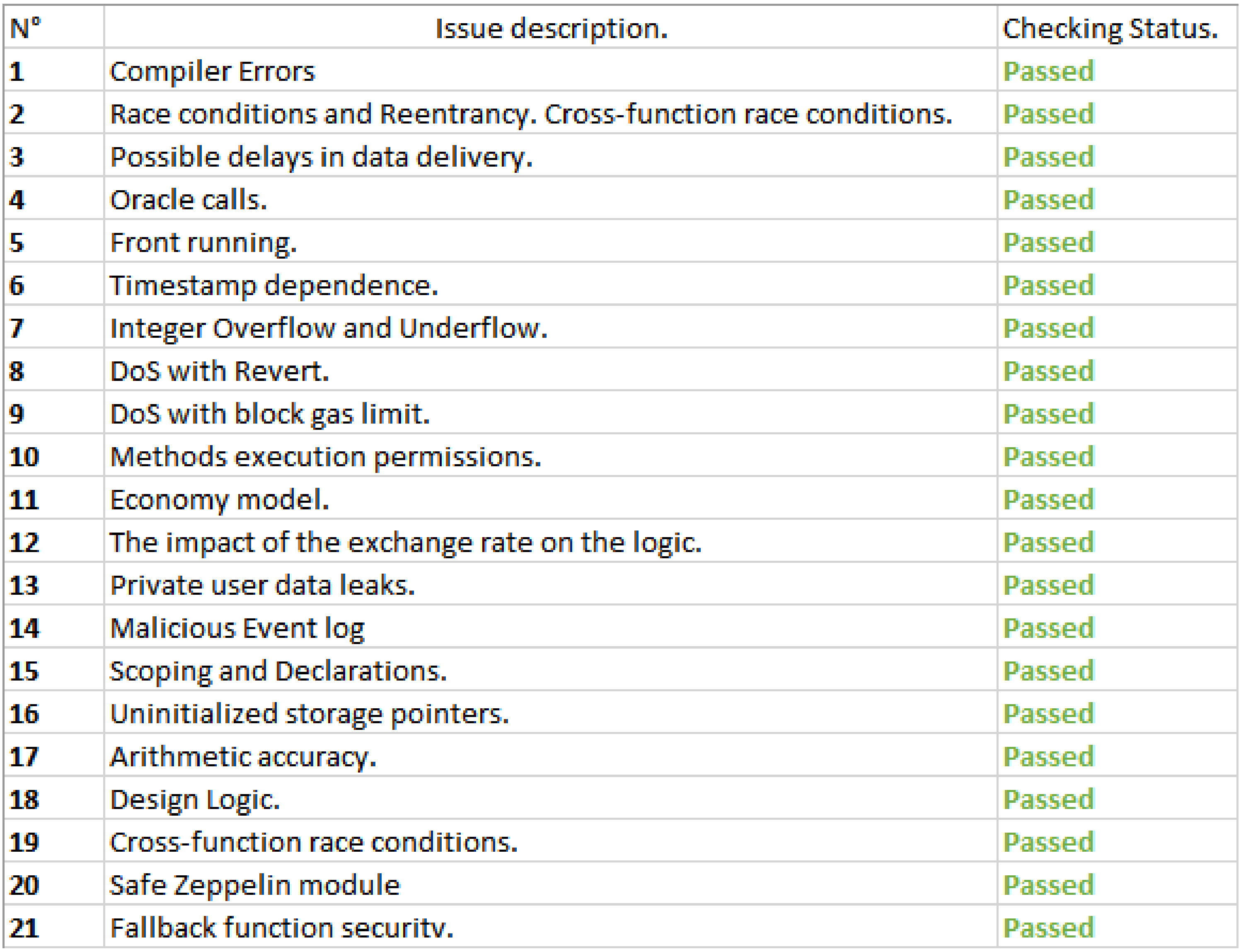
Owner Privileges:



```
     * Requirements:
     *
     * - the calling contract must have an ETH balance of at least `value`.
     * - the called Solidity function must be `payable`.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value
    ) internal returns (bytes memory) {
        return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
    }

    /**
     * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
     * with `errorMessage` as a fallback revert reason when `target` reverts.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value,
        string memory errorMessage
    ) internal returns (bytes memory) {
        require(address(this).balance >= value, "Address: insufficient balance for call");
        require(isContract(target), "Address: call to non-contract");

        (bool success, bytes memory returndata) = target.call{value: value}(data);
```

```
function renounceOwnership() public virtual onlyOwner {
function transferOwnership(address newOwner) public virtual onlyOwner {
function distributeSOLANADividends(uint256 amount) public onlyOwner{
function excludeFromDividends(address account) external onlyOwner {
function updateClaimWait(uint256 newClaimWait) external onlyOwner {
function setBalance(address payable account, uint256 newBalance) external onlyOwner {
function processAccount(address payable account, bool automatic) public onlyOwner returns (bool) {
function updateDividendTracker(address newAddress) public onlyOwner {
function updateUniswapV2Router(address newAddress) public onlyOwner {
function excludeFromFees(address account, bool excluded) public onlyOwner {
function excludeMultipleAccountsFromFees(address[] calldata accounts, bool excluded) public onlyOwner {
function setDevWallet(address payable wallet) external onlyOwner{
function setFoudWalletAddress(address payable wallet) external onlyOwner{
function setFee(
function setAutomatedMarketMakerPair(address pair, bool value) public onlyOwner {
function blacklistAddress(address account, bool value) external onlyOwner{
function snipe(address bot)onlyOwner() public{
function rescueToken(address tokenAddress, uint256 tokens)
function rescueBNB(address payable _recipient) public onlyOwner {
function updateGasForProcessing(uint256 newValue) public onlyOwner {
function updateClaimWait(uint256 claimWait) external onlyOwner {
function excludeFromDividends(address account) external onlyOwner{
function setSnipeBlocks(uint8 _blocks) external onlyOwner {
function setSnipeTime(uint256 _snipeTime) external onlyOwner {
```

Issue Checking:

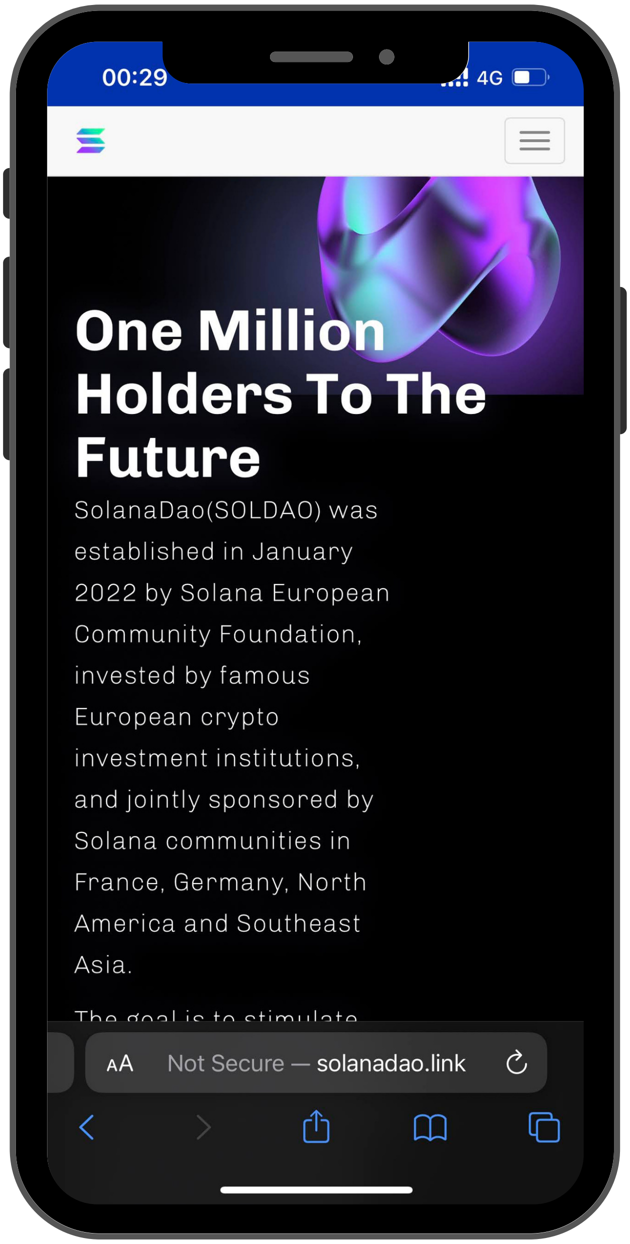


N°	Issue description.	Checking Status.
1	Compiler Errors	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Passed
19	Cross-function race conditions.	Passed
20	Safe Zeppelin module	Passed
21	Fallback function securitv.	Passed

Audit Result:

PASSED

WEBSITE - OVERVIEW



One Million Holders To The Future

SolanaDao(SOLDAO) was established in January 2022 by Solana European Community Foundation, invested by famous European crypto investment institutions, and jointly sponsored by Solana communities in France, Germany, North America and Southeast Asia.

The goal is to stimulate

🕒 Domain Registration: 2022-04-11

8 days old

SSL CERTIFICATE: A

(https://www.ssllabs.com)

SOCIAL MEDIA - OVERVIEW

SOLANADAOOFFICIAL

SOLANADAOTOKEN

SOLANADAO.LINK